App. No. 09/895,449
Amdt. dated June 10, 2004
Reply to Office Action of February 10, 2004

1.   (Original)   A computer-implemented method for removing an object from a namespace, comprising:

receiving a notification that an unload operation is starting, the unload notification identifying the object to be unloaded from the namespace;

in response to the unload notification, detaching the identified object from the namespace while maintaining the identified object in a location in memory;

identifying a device associated with the identified object as invalid;

informing an operating system component that the device associated with the identified object is invalid; and

in response to a notification that the associated device is no longer being used, causing the identified object to be freed from the location in memory.

2.   (Original)   The computer-implemented method of claim 1, further comprising:

prior to detaching the identified object, suspending power management and configuration management activity to put the namespace in a steady state; and

after the identified object is detached from the namespace, resuming power management and configuration management activity.

3.   (Original)   The computer-implemented method of claim 1, further comprising:

prior to detaching the identified object, locking access to the namespace; and

after the identified object is detached from the namespace, unlocking access to the namespace.

4.   (Original)   The computer-implemented method of claim 1, wherein the unload notification indicates that a hardware device associated with the object is being made unavailable.

5.   (Original)   The computer-implemented method of claim 4, wherein the unload notification comprises means for indicating that the hardware device is being made unavailable.

6.    (Original)    The computer-implemented method of claim 5, wherein the means for indicating that the hardware device is being made unavailable comprises an interrupt.

7.    (Original)    The computer-implemented method of claim 6, wherein the interrupt comprises a general purpose event signal.

8.    (Currently Amended) The computer-implemented method of claim 7 [[4]], wherein the general purpose event signal is generated by the device associated with the identified object.

9.    (Original)    The computer-implemented method of claim 1, wherein detaching the identified object comprises severing a link between the identified object and the namespace and flagging another object in the namespace to indicate that the identified object has been severed.

10.   (Original)    The computer-implemented method of claim 9, wherein maintaining the identified object in a location in memory comprises maintaining a pointer to the location in memory in a device extension associated with the device.

11.   (Original)    The computer-implemented method of claim 1, wherein the notification that the associated device is no longer being used comprises a notification that a reference count associated with the identified object has passed a predetermined threshold.

12.   (Original)    The computer-implemented method of claim 11, wherein the reference count is maintained in a device extension associated with the device, and wherein the predetermined threshold indicates that objects no longer refer to the device.

13.   (Original)    A computer-readable medium having computer-executable instructions for unloading an object from a namespace, comprising:

in response to receiving a notification of an unload operation, identifying the object within the namespace as being unloaded;

detaching the object from the namespace while maintaining the object in a memory location;

informing a driver responsible for coordinating access to the object that the object has been detached from the namespace; and

in response to a notification from the driver that the object is no longer being used, deleting the object from the memory location.

14.    (Original)    The computer-readable medium of claim 13, further comprising:

prior to detaching the object from the namespace, notifying the driver that the unload operation is occurring and identifying the object to the driver.

15.    (Original)    The computer-readable medium of claim 13, wherein identifying the object within the namespace as being unloaded comprises setting a flag associated with the object that indicates that the object is being unloaded.

16.    (Original)    The computer-readable medium of claim 15, further comprising setting a second flag associated with a parent object of the object, the second flag indicating that the parent object includes invalid references.

17.    (Original)    The computer-readable medium of claim 13, wherein detaching the object from the namespace comprises severing a link between the object and a parent object of the object.

18.    (Original)    The computer-readable medium of claim 13, further comprising, prior to informing the driver that the object has been detached, modifying the object such that a method associated with the object may not be executed.

App. No. 09/895,449
Amdt. dated June 10, 2004
Reply to Office Action of February 10, 2004

19.    (Original)    A computer-readable medium having computer-executable instructions for unloading an object from a namespace, comprising:

receiving a notification that an unload operation of a device is beginning;

in response to the notification, performing preprocessing operations to prepare a configuration management system for the unload operation;

receiving a notification that an object has been detached from the namespace, the object being associated with the device;

informing an operating system component that the unload operation is beginning and identifying the device;

awaiting a notification that the object is no longer being used by the operating system; and

issuing a notification to the configuration management system that the object is no longer being used.

20.    (Currently Amended) The computer-readable medium of claim 19, wherein the preprocessing operations includes invalidating a general purpose event ~~even~~ mask associated with the configuration management system.

21.    (Original)    The computer-readable medium of claim 20, wherein the computer-executable instructions further comprise, after receiving the notification that the object is no longer being used by the operating system, rebuilding the general purpose event mask associated with the configuration management system.

22.    (Original)    The computer-readable medium of claim 19, wherein the notification that the object is no longer being used comprises a reference count value associated with a number of resources that refer to a device extension associated with the device.

23.    (Original)    The computer-readable medium of claim 22, wherein the reference count value indicates that only one resource refers to the device extension and the one resource comprises the object being unloaded.

24.     (Original)     The computer-readable medium of claim 19, further comprising:

receiving from the operating system a message intended for the detached object; and

directing the message to the detached object by retrieving a pointer to the detached object from a data structure maintained by the operating system associated with the device.

25.     (Currently Amended) A computer-readable medium having computer-executable components, comprising:

an AML interpreter configured to maintain a namespace of objects associated with hardware devices, and to detach an object from the namespace in response to an unload message, the object being associated with a device, the AML interpreter being further configured to issue a notification that the object has been detached from the namespace; and

an ACPI driver configured to receive the notification from the AML interpreter and to cause a device object associated with the device to be unloaded, and to issue a notification to the AML interpreter when the device object is successfully unloaded.

26.     (Original)     The computer-readable medium of claim 25, wherein the ACPI driver is further configured to interface with a device extension associated with the device object, the device extension including a reference counter indicative of a number of resources that refer to the device object.

27.     (Original)     The computer-readable medium of claim 26, wherein the ACPI driver is further configured to issue the notification to the AML interpreter in response to the device extension indicating that resources no longer refer to the device object.

28.     (Original)     The computer-readable medium of claim 27, wherein the AML interpreter is further configured to delete the detached object from memory in response to the notification from the ACPI driver.

29.     (Original)     The computer-readable medium of claim 25, wherein the ACPI driver is further configured to direct a message to the detached object.

30.    (New) A computer-implemented method for unloading a device from a
namespace, comprising:

generating an unload event, wherein the unload event indicates that the device is to be
removed from the namespace;

informing a driver that the unload event has been generated;

identifying a namespace object affected by the unload event, wherein the namespace
object is associated with the device;

sending the identified namespace object to the driver;

detaching the identified namespace object from the namespace while maintaining the
identified object in a memory location;

informing the driver that the unload event has been completed; and

eliminating the identified namespace object from the memory location.


31.    (New) A computer-implemented method for unloading a device from a
namespace, comprising:

initiating an unload event, wherein the unload event indicates that the device is to be
removed from the namespace;

receiving a notification that the unload event has been initiated;

identifying a namespace object affected by the unload event, wherein the namespace
object is associated with the device;

removing the identified namespace object from the namespace;

storing the identified namespace object in a memory location; and

notifying an interpreter that the identified namespace object has been removed from the
namespace.